# Communication Needs Assessment for Distributed Turbine Engine Control

*Dennis E. Culley*
*Glenn Research Center, Cleveland, Ohio*

*Alireza R. Behbahani*
*Wright-Patterson Air Force Base, Dayton, Ohio*

# NASA STI Program . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI Program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NASA Aeronautics and Space Database and its public interface, the NASA Technical Reports Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include creating custom thesauri, building customized databases, organizing and publishing research results.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at *http://www.sti.nasa.gov*

- E-mail your question via the Internet to *help@ sti.nasa.gov*

- Fax your question to the NASA STI Help Desk at 301–621–0134

- Telephone the NASA STI Help Desk at 301–621–0390

- Write to:
  NASA Center for AeroSpace Information (CASI)
  7115 Standard Drive
  Hanover, MD 21076–1320

# Communication Needs Assessment for Distributed Turbine Engine Control

*Dennis E. Culley*
*Glenn Research Center, Cleveland, Ohio*

*Alireza R. Behbahani*
*Wright-Patterson Air Force Base, Dayton, Ohio*

National Aeronautics and
Space Administration

Glenn Research Center
Cleveland, Ohio 44135

September 2008

# Communication Needs Assessment for Distributed Turbine Engine Control

Dennis E. Culley
National Aeronautics and Space Administration
Glenn Research Center
Cleveland, Ohio 44135

Alireza R. Behbahani
Air Force Research Laboratory
Wright-Patterson Air Force Base
Dayton, Ohio 45433

## Abstract

Control system architecture is a major contributor to future propulsion engine performance enhancement and life cycle cost reduction. The control system architecture can be a means to effect net weight reduction in future engine systems, provide a streamlined approach to system design and implementation, and enable new opportunities for performance optimization and increased awareness about system health. The transition from a centralized, point-to-point analog control topology to a modular, networked, distributed system is paramount to extracting these system improvements. However, distributed engine control systems are only possible through the successful design and implementation of a suitable communication system. In a networked system, understanding the data flow between control elements is a fundamental requirement for specifying the communication architecture which, itself, is dependent on the functional capability of electronics in the engine environment. This paper presents an assessment of the communication needs for distributed control using strawman designs and relates how system design decisions relate to overall goals as we progress from the baseline centralized architecture, through partially distributed and fully distributed control systems.

## I. Introduction

The realization of true distributed control in aeropropulsion engine systems has been a long awaited development. Industry and government efforts have been in place since the late 1980s to develop the necessary underlying technologies, especially in the area of high temperature electronics. Much of this work has been, and continues to be, performed within the larger objectives of the Integrated High Performance Turbine Engine Technology (IHPTET) initiative and its successor the Versatile Affordable Advanced Turbine Engine (VAATE) plan. While the technology appears to be within reach, there still remain some fundamental decisions about the architecture of such systems and how the industry could, or even should, cooperate in their development.

There is no specific methodology or example for distributed engine control in the aeropropulsion community although there is movement in that direction for segments of small subsystems on development programs. Part of the dilemma faced by end-users, engine manufacturers, and suppliers is the lack of clearly defined benefits for what amounts to a major change in systems development methodology and how those changes will affect stakeholder interactions. The Distributed Engine Control Working Group (DECWG) is a consortium of government and industry stakeholders in aeropropulsion systems. It was formed to explore these issues and to attempt to understand and quantify the effects of these changes.

As participants in the DECWG, it is clear to the authors that an opportunity exists to help coordinate industry efforts on the grounds of commonality in order to extract the maximum benefit for every

participant from supplier to end-user. The process control industries (and more recently the automotive industry (ref. 1)) have moved into distributed control systems as the norm. Aerospace is slowly moving in this direction as well but will be at a disadvantage with respect to other industries due to its relatively small market size and severe system requirements imposed by the operating environment and reliability needs. The lack of influence on the electronic component supplier market, on which it critically depends, is a major factor in system cost and an important reason for industry-wide cooperation. In distributed systems this cooperation can best be expressed in the form of identification and/or development of open systems standards, especially those applied at the subsystem and component interfaces within the engine control system. These standards, such as the IEEE–1451 standard for a smart transducer interface for sensors and actuators (ref. 2), can broaden the influence of the aerospace community on electronics markets and lead to other system-wide benefits in terms of performance and innovative new capabilities.

The IEEE–1451 standard helps describe what information is communicated in a distributed system, whereas how it is communicated will be defined by other standards. Together these define the communication network which is the critical interface which ties the components and subsystems together to form a unified whole. One objective of this effort is to assess and describe the function, interaction, and necessity of major communication networks in modular, distributed engine control. This perspective is different from one which considers how existing communication networks might fit the engine application (ref. 3).

The overall objective of this paper is to provide an independent assessment of the communication needs for the distributed turbine engine control system application. The paper is organized in sections to gradually build up the logic behind the assessment. It begins by briefly framing the underlying objectives for changing the control architecture; it is critically important to maintain a focus on these objectives when evaluating various aspects of the control architecture. This is followed by the description of a representative control system, implemented in a centralized architecture, which is subsequently used as a baseline for comparison. The same system of engine sensors and actuators is then conceptually described as a partially distributed system (meaning a network of smaller control nodes employing analog input/output) and finally as a fully distributed system (where each system element communicates over the digital network). The communication needs are compared for each configuration, thereby providing a vehicle for industry to discuss the possibility of converging on a common, standard approach to communications for distributed controls.

# II. The Rationale for Distributed Control System Architecture

The overall benefit of the transition to a modular, distributed engine control architecture is realized through a combination of weight reduction, life-cycle cost reductions and flexibility in future system design/redesign. These topics are thoroughly discussed in separate papers by Culley (ref. 4), et al., and by Behbahani (ref. 5), et al., the latter enjoying broad participation by the DECWG. Below, these topics are briefly provided to the reader to aid comprehension of the overall objectives and to provide context for the communication systems assessment that begins in section III.

## A. Weight Reduction

It must be understood that a feasible distributed control system will only result when it can be demonstrated that it provides the same level of performance as current state-of-the-art engine control systems. A loss of engine performance, such as a decrease in thrust or an increase in weight, will not be tolerated in real world systems in which small performance margins equate to survivability in hostile combat situations or tens of millions of dollars in annual revenue for commercial aviation. A simple assessment of control system weight can be achieved by considering the three component classes of control system hardware, first independently and then as a system. These are the Full Authority Digital Engine Controller (FADEC), the system sensors and actuators, and the wiring harness which

interconnects the previous two component classes. While it is difficult to establish precise weight estimates due to the variability of system designs, the elementary logic is clear.

The FADEC is a data acquisition and high performance control law processing system implemented in an environmentally hardened avionics package. In a centralized architecture, the FADEC is responsible for implementing each I/O channel interfacing to every system sensor and actuator. This is done redundantly to address reliability issues and minimize failure effects. The amount of I/O varies with engine size and the complexity of the control; however, it can be assumed that the customized signal conditioning electronics consumes approximately 50 percent of the package volume. Data processing and gate array logic circuitry, which is driven by technology developments in the mainstream electronics industry, have seen an exponential increase in density. This has resulted in a smaller volume requirement for high performance processing electronics such as those used for FADEC control law processing. Thus, in aero-propulsion, the trend toward more complex control systems, with increased numbers of system effectors, will continue to accentuate the volume proportion dedicated to I/O interfacing circuitry. A distributed control system eliminates this circuitry dedicated to analog I/O signaling within the FADEC and replaces it with a shared digital network interface. All things being equal, a weight and volume reduction of the FADEC avionics assembly is clearly achieved through distributed architecture.

Sensors and actuators are fixed elements in the engine control system. Their size cannot be reduced and their location cannot be altered simply by a transition of the control system architecture. In fact, the volume of these elements must increase under distributed control, due to the added burden of signal conditioning and common networked communications imposed by the new FADEC interface. All things being equal, the volume and weight lost from the FADEC signal conditioning electronics is likely to be regained by the system sensors and actuators for a net increase in weight and volume for these system elements.

In fact, it is the wiring harness weight that tips the scales in favor of distributed control. The large, bulky, and complex wiring bundle needed to implement point-to-point analog signaling between the FADEC and each system element is greatly reduced through the utilization of digital serial communication. The reduction is achieved through significantly fewer wires, a reduction in the diameter of the shielding, and perhaps an overall reduction in total length. Since the harness weight is large with respect to electronics weight this reduction should be substantial. The overall net effect of distributed control architecture, considering the FADEC, system effectors, and wire harness is likely to be a reduction in total system weight.

The preceding assessment makes the assumption that the electronic components embedded in system effectors are capable of implementing an appropriate communication protocol and surviving the environment at that location. System designers may choose, or be forced to choose, weight penalties for thermal control to insure electronics survivability. As long as there is no net weight increase between centralized and distributed systems there is no loss of overall engine system performance related to this aspect of control architecture. Over the long term, technology and innovation favors the distributed approach because of harness weight and the increasing need for control complexity in engine applications.

## B. Life-Cycle Cost Reduction

Engine systems have a long lifespan with respect to the components of which they are comprised. The rising complexity of engine systems is a driving influence behind increasing design cycle times and the inevitable redesign due to changing system requirements and issues related to electronic component obsolescence. Distributed systems, by their nature, fundamentally address this issue by forcing a decomposition of a single complex system into smaller, less complex subsystems interconnected by well-defined interfaces. When properly designed, distributed systems effectively segregate and firewall system functions at the interface. This leads to a modular approach to system design, encouraging subsystem reuse and limiting the scope of redesign efforts undertaken during the engine system lifespan.

One issue plaguing engine control system designers is the lack of availability and selection of electronic components suitable for high reliability, mission critical systems in harsh environments. Insufficient profitability due to the size of this market segment discourages electronics manufacturers from supplying these products. This has forced system integrators to rely on environmentally screened commercial products or custom packaged modules acquired at great cost. Even more significant is the limited duration each commercial product is available since the life expectancy of electronic components is driven by consumer demand for increasingly higher performance. In a centralized engine control architecture these obsolescence issues tend to ripple throughout the control system forcing extensive redesign and costly requalification. Distributed control addresses this issue in two ways. First, it may increase the market size through industry participation in open system standards. Second, it can isolate the effect of obsolescence through the functional segregation of subsystem interfaces, allowing alternative subsystem designs to provide the equivalent functionality at minimal impact to the larger system.

There is a great deal of commonality in the control of every turbine engine system. However, specific implementations vary from one manufacturer to another and their intellectual property is closely guarded. Even so, sensed parameters, actuation requirements, processing capabilities, and data flow are more similar than they are different. These areas of commonality can be exploited through modular system design, resulting in significant reduction of non-recurring engineering costs for suppliers and manufacturers. Resources once focused on the creation of single point component designs can now be redirected toward the creation of value-added functionality within system components, by virtue of embedded intelligence, thereby encouraging competition based on price and product differentiation.

With the embedded intelligence necessary for a networked control system, much of the product differentiation of system components will revolve around maintenance issues related to system availability, mission success, and performance because these capabilities add value by reducing operational costs for the end user. Innovation within the functional system elements can be directed toward simplifying, adjusting, and maintaining calibration; increasing built-in test capability; and providing fault isolation at the line replaceable unit. Of course, improving the weight and volume of system sensors and actuators through improved packaging and integration would address issues raised in the previous section.

As with the weight reduction assessment, electronics which can survive in an embedded environment on a turbine engine are also assumed necessary to the assessment supporting life cycle cost reduction. Another challenge will be developing the regulatory infrastructure to allow subsystem requalification in order to enable the full exploitation of the benefits of distributed systems in relation to obsolescence and modular subsystems. Finally, acquisition cost for end-users may increase to support the added value of these intelligent systems. Much of this will undoubtedly come from the cost of extended temperature range electronics which are not in demand by the consumer market. The modularity of a distributed control system architecture, using standardized, reusable electronics should none-the-less contribute to reduced life-cycle costs.


## C. Future System Design Flexibility

Control system architecture is an enabling technology for future adaptive engine systems which will employ controls in ways which do not easily integrate into an existing centralized architecture. These capabilities, such as active component control, may require extremely fast response times and/or processing capabilities not typically used in present systems. The ultimate goal of modular, distributed control architecture is to provide the capability for a flexible and scalable control system design. This capability can lead to highly customizable propulsion systems tailored to specific customer needs. The capability can also extend to airframe control, blurring the delineation between the engine and the airframe, leading to highly integrated air vehicle systems with improved performance capabilities.

# III. A Baseline Centralized Engine Control System

A centralized control architecture is typically characterized by a single, engine-mounted FADEC which connects directly to each control system element. The environmentally controlled electronics package of the FADEC insures the survivability of all the circuitry necessary for sensor and actuator operations and control law processing. A description of this system is provided to establish a baseline for consideration of the impact of a distributed architecture on data flow and communications.

Figure 1 depicts a generic engine control system hardware connection diagram. Shown is a centralized FADEC with a suite of sensors used for control, a suite of auxiliary sensors used for monitoring engine condition, and a set of actuators that enable safe and reliable engine operation over a range of operating conditions and in response to the pilot throttle command. Figure 2 shows the station map of the engine system with approximate gas path temperatures at several locations. The gas path temperatures can be used to estimate the temperature environment at the various control element locations identified by the element's subscripted station number and the description provided in table I. The basic sensed parameters in an engine control system fall into the categories of temperature, pressure, speed, flow, and position. Each specific sensor is selected based on the required range and precision of the control or health monitoring application.
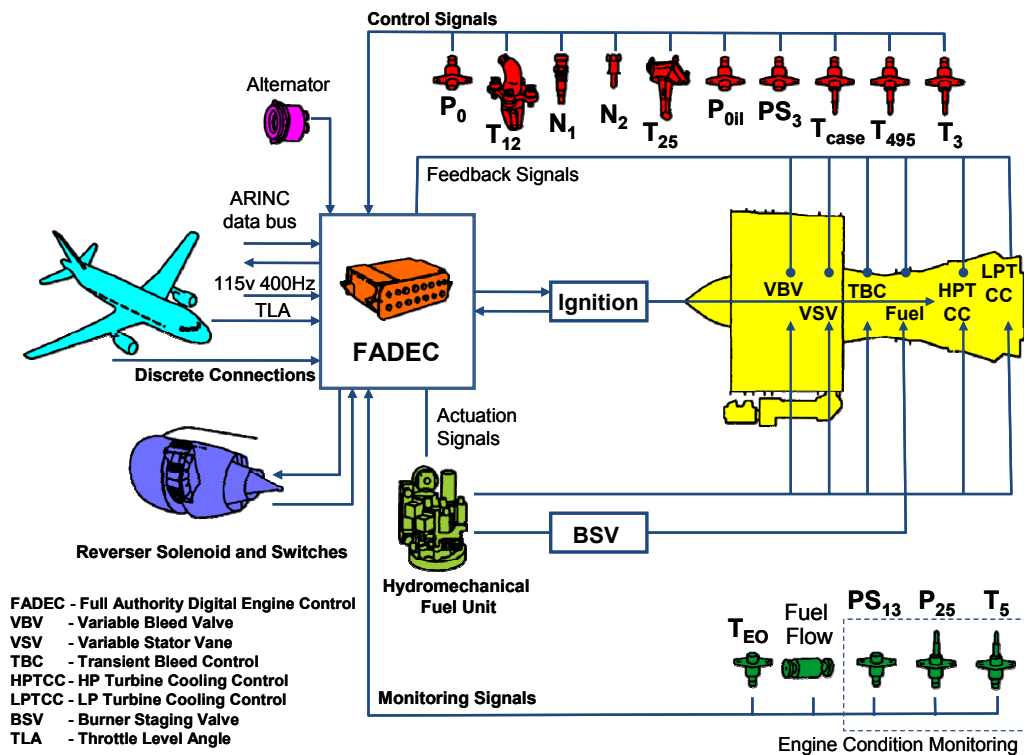


Figure 1.—Baseline centralized engine control architecture. The FADEC connects directly to each system element.
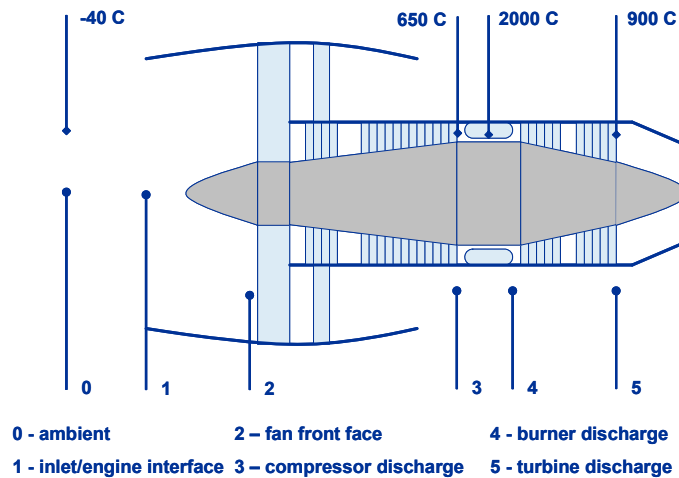
Figure 2.—Turbine engine station diagram with approximate gas path temperatures.

TABLE I.—SENSORS AND ACTUATORS OF THE BASELINE CENTRALIZED SYSTEM
[Update rates marked with an asterisk are supplied by Volponi in reference 6.]

| Symbol | Function | Update Rate Hz | Bit Length | bit rate |
|---|---|---|---|---|
| $P_0$ | ambient total pressure | 5* | 16 | 80 |
| $T_{12}$ | total temperature fan inlet | 5* | 16 | 80 |
| $N_1$ | LP spool speed | 20* | 16 | 320 |
| $N_2$ | HP spool speed | 20* | 16 | 320 |
| $T_{25}$ | total temperature HPC inlet | 20* | 16 | 320 |
| $P_{oil}$ | engine oil pressure | 5 | 16 | 80 |
| $PS_3$ | static pressure compressor discharge | 5* | 16 | 80 |
| $T_{case}$ | case temperature | 5 | 16 | 80 |
| $T_{495}$ | total temperature LPT inlet | 20 | 16 | 320 |
| $T_3$ | total temperature compressor discharge | 5* | 16 | 80 |
| $T_{E0}$ | engine oil temperature | 5 | 16 | 80 |
| Fuel Flow | fuel flow meter | 5* | 16 | 80 |
| $PS_{13}$ | static pressure fan discharge | 5 | 16 | 80 |
| $P_{25}$ | total pressure HPC inlet | 20* | 16 | 320 |
| $T_5$ | total temperature turbine discharge | 20* | 16 | 320 |
| VBV | variable bleed valve | 5 | 32 | 160 |
| VSV | variable stator vanes | 5 | 32 | 160 |
| TBC | transient bleed control | 5 | 32 | 160 |
| Fuel | fuel valve | 5 | 32 | 160 |
| HPTCC | HPT cooling control valve | 5 | 32 | 160 |
| LPTCC | LPT cooling control valve | 5 | 32 | 160 |
| Ignition | ignitor | 5 | 32 | 160 |
| Thrust Reverser | thrust reverser | 5 | 32 | 160 |
| Solenoids | various control isolation valves | 5 | 32 | 160 |
| Average bit rate | | | | 4080 |
| Max. effective bit rate at 20 Hz ( 50 msec interval) | | | 528 | 10560 |
| Max. effective bit rate at 100 Hz rate ( 10 msec interal) | | | | 52800 |

In a typical FADEC, the input channel dedicated to each sensor provides the signal conditioning circuitry to power the device, and then buffer, filter, and scale the incoming signal. The signal is then digitized at an appropriate rate determined by factors such as the sensor time constant, software filtering requirements, and the update rate of the control laws that require the sensor data. Once the data at the input channel is digitized it becomes available in system memory for further manipulation and to be operated on by the system processor which implements the various control laws. Many of these same digitized values are also monitored by programmable hardware circuits for fast rate of change and limit checks.

On the output side, FADEC channels are used to command system actuators. These commands are the result of the hardware and software implementation of the control laws, and are subject to the temporal constraints of the input data. The digital commands are converted to the analog domain, typically by a D/A converter circuit, where they are then amplified to meet the power requirements of the actuator. The required response times from the actuators are determined by the time constants of the turbomachinery and are typically much faster than the process being controlled. The commands to the actuators from the FADEC must satisfy these requirements.

To understand the impact of architecture transformation on control system data flow, it is important to estimate an equivalent data rate for the baseline centralized system. Listed in table I are the symbols of the control elements from figure 1 and a description of their functions. The update rates of the system elements listed are based on a combination of data from Volponi (ref. 6) and the author's own estimates. In a centralized FADEC, the input and output values are stored as integer values resulting from the A/D conversion process (input) or driving the D/A conversion process (output). Based on a reasonable precision for such information they are assumed to be 12-b integer values corresponding to a resolution of 0.02 percent of full scale. Since digital values are typically stored as bytes (8-b units) it is appropriate to round each I/O value to a 16-b word (2 B) to accommodate signed values or increased precision. Assuming the data flows over a common serial media, the equivalent total data flow of the analog FADEC I/O system is simply the sum of products of the data size and update rate for each I/O channel. Note that actuators are assumed to have feedback position sensors that are not shown in the table. Since they receive and transmit data their assigned bit length is doubled to 32 b. For the I/O channels and data rates listed in table I, these result in an effective bit rate of 4080 bps.

In real-time control the temporal relationship between data is an important consideration. Many hardware data acquisition systems employ simultaneous sampling of the analog channels to insure that all data is acquired without introducing phase lag during the digitization process. The effect this has on data flow in the centralized system can be estimated by assuming the entire transmission of data from all channels occurs within the span of the fastest update interval of the system. From Table I this update rate is 20 Hz and, therefore, the effective bit rate must consider that every datum is transmitted within this update interval even if that datum is used at lower update rate. This increases the maximum effective bit rate to 10,560 bps. If the maximum FADEC update rate were to increase to 100 Hz, corresponding to a 10 msec control interval as suggested by Soeder (ref. 7), this increases the maximum bit rate to 52,800 bps. The data rates given assume that the I/O data transfer is an independent process which does not affect control law processing time.

These bit rates may not tell the entire story about the effective data rate at the control law processor. Every analog signal is susceptible to spurious noise introduced from the environment. To counteract this reality, low-pass filtering of the analog channel is used to limit the highest frequency that can be captured by the analog system. This is then followed by a digital filter which averages a series of acquired values, further limiting the effective frequency of the digitized signal used in the calculation of control values. This is depicted in the I/O block diagram of figure 3, which shows a single input and single output channel used for control. It is clear that where one draws the interface between the process blocks in the control system can greatly affect the resulting data flow in a networked version of the control system. The estimated bit rates described above can be used to bound the problem.
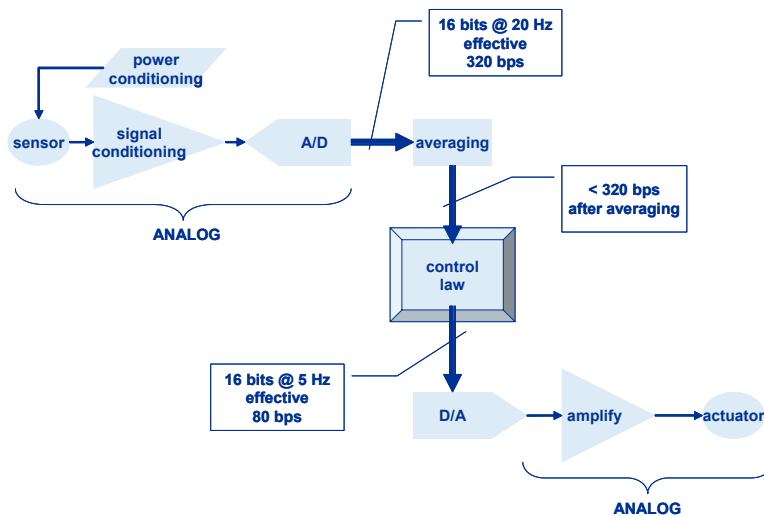
Figure 3.—Block diagram of analog I/O channels.

In an engine control architecture there will be a strong emphasis on optimization of both the hardware and software to minimize weight and cost. Software optimization, however, may place emphasis on speed of execution and minimization of system resources in lieu of easily understood code and portability. In a centralized architecture, understanding and portability issues do not predominate because it is a closed system. These optimization issues will also be shown to affect communications and data flow because the use of a standardized data structure will require a greater number of bits than the native binary representation.

# IV. Partially Distributed Control System

A partially distributed control system is characterized by a network of smaller electronic enclosures which implement all the functions of engine control between them. The network is coordinated by a supervisory FADEC which may, or may not be engine-mounted. The local control nodes are similar to the centralized FADEC in that they perform input/output operations and closed-loop control. However, the scope of the local control node's ability to control engine functions is limited to the suite of control elements immediately connected to it. Loop closures involving control elements from multiple local control nodes are assumed to be closed through the supervisory FADEC.

## A. Partially Distributed Strawman Design

The centralized control system presented in section III is transformed into a partially distributed design which functions as described in the preceding paragraph. This strawman system is illustrated in figure 4. The design is based on four primary criteria:

1. Eliminate I/O functions other than networking from the FADEC
2. Minimize wire harness length (and therefore weight)
3. Evenly distribute the I/O load across the local nodes
4. Require at least one actuator function at each local node

A total of four local controllers are chosen to implement the on-engine components; identified as the inlet/fan node, compressor node, combustor node, and turbine/nozzle node. This design is somewhat arbitrary, but is intended to illustrate the impact of inter-node networked communications. Many other configurations and optimization criteria are possible.
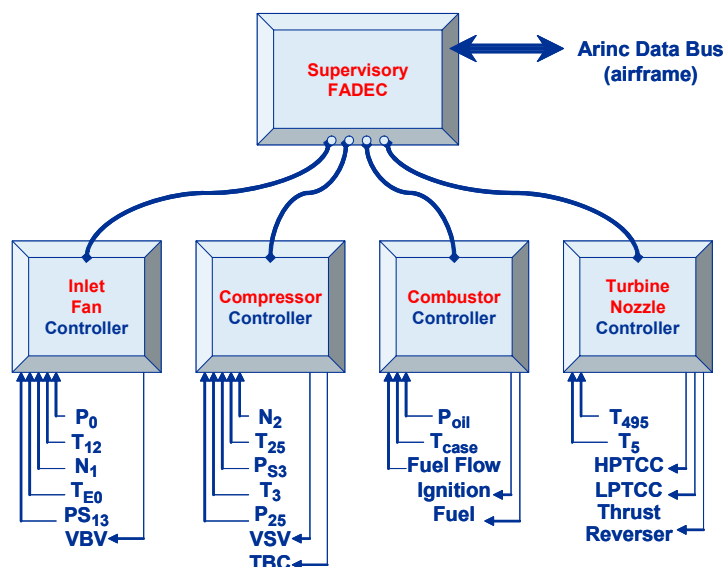
Figure 4.—Partially distributed engine control strawman configuration generated from baseline system in figure 1.

## B. Loop Closure in Partially Distributed Systems

In the centralized system, all control law processing and loop closure is performed in one location. In a distributed system, we must begin to understand the complexities of these control laws to determine how loop closure occurs and determine the feasibility of distributing control law functionality.

The fundamental control of a propulsion engine is based on producing a constant thrust which is proportional to the commanded throttle position. Early engine controls used simple techniques to estimate engine power and regulate the fuel flow. Since that time, engine performance has continued to dramatically improve as new materials, structures, aerothermodynamic technologies, and abilities to accurately control them have become more refined. Although controlling to a constant thrust remains the fundamental purpose of engine control, keeping the system within safe operating limits, offloading pilot burden, optimizing system performance, and a host of diagnostic and health monitoring functions are perhaps the larger share of modern control requirements.

Engine thrust is sensitive to a variety of factors including; engine rpm, nozzle area, fuel flow, compressor bleed, turbine temperature, airspeed, and ambient temperature, pressure, and humidity. An excellent discussion of these effects is provided by Treager (ref. 8) A simple review of these parameters, which correspond to various sensor and actuator elements shown on the centralized system diagram of figure 1, reveals that it would be nearly impossible to group together the required control elements to carry out closed loop control within one local control module in a partially distributed system. In fact, that approach defeats the purpose of using distributed control because it reduces the designer's flexibility and the performance and cost reduction benefits of the architecture.

The implication of this reality, under our definition of a partially distributed control system, is that very little closed-loop control occurs at the remote nodes. Most local closed-loop control would simply involve verification of actuator commands.

## C. Network Implementation

In a partially distributed system the data flow must replicate the effective data flow that was described in the baseline centralized system. The simplest implementation would be to install separate, point-to-point serial communication channels between each local control node and the FADEC. The main advantages of this approach are the simplicity of the hardware and little need for an elaborate

communication protocol. Separate communication channels between the FADEC and the local node eliminate the possibility of simultaneous data transmission (data collision) on the media and ambiguity about the source and destination of the messages. Synchronization would be controlled by the FADEC. The addition of forward error correction (FEC) would slightly increase the amount of information to be transmitted. Forward error correction is a technique which enables the receiving device to correct corrupted data without requiring retransmission. The estimated data flow rates for individual communication channels between the FADEC and each local node are shown in table II. Only information which replicates the function of the centralized data flow is shown. The communication over the individual serial channels occurs in both directions and is shown for maximum update rates of 20 Hz (50 msec interval) and 100 Hz (10 msec interval).

TABLE II.—ESTIMATED EFFECTIVE BIT RATE FOR POINT-TO-POINT
COMMUNICATION BETWEEN FADEC AND LOCAL CONTROL NODES

| Symbol | I/O Update Rate Hz | Inlet/Fan bits | | Compressor bits | | Combustor bits | | Turbine/ Nozzle bits | |
|---|---|---|---|---|---|---|---|---|---|
| | | in | out | in | out | in | out | in | out |
| $P_0$ | 5 | | | | | | | | |
| $T_{12}$ | 5 | | 16 | | | | | | |
| $N_1$ | 20 | | 16 | | | | | | |
| $N_2$ | 20 | | | | 16 | | | | |
| $T_{25}$ | 20 | | | | 16 | | | | |
| $P_{oil}$ | 5 | | | | | | 16 | | |
| $PS_3$ | 5 | | | | 16 | | | | |
| $T_{case}$ | 5 | | | | | | 16 | | |
| $T_{495}$ | 20 | | | | | | | | 16 |
| $T_3$ | 5 | | | | 16 | | | | |
| $T_{E0}$ | 5 | | 16 | | | | | | |
| Fuel Flow | 5 | | | | | | 16 | | |
| $PS_{13}$ | 5 | | 16 | | | | | | |
| $P_{25}$ | 20 | | | | 16 | | | | |
| $T_5$ | 20 | | | | | | | | 16 |
| VBV | 5 | 16 | 16 | | | | | | |
| VSV | 5 | 16 | 16 | 16 | 16 | | | | |
| TBC | 5 | | | 16 | 16 | | | | |
| Fuel | 5 | | | | | 16 | 16 | | |
| HPTCC | 5 | | | | | | | 16 | 16 |
| LPTCC | 5 | | | | | | | 16 | 16 |
| Ignition | 5 | | | | | 16 | 16 | | |
| Thrust Reverser | 5 | | | | | | | 16 | 16 |
| Solenoids | 5 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 16 |
| FEC bits per message | | 7 | 8 | 7 | 9 | 7 | 8 | 8 | 8 |
| Total bits per message | | 55 | 120 | 55 | 137 | 55 | 104 | 72 | 104 |
| Max. effective bit rate 20 Hz ( 50 msec interval) | | 3500 | | 3840 | | 3180 | | 3520 | |
| Max. effective bit rate 100 Hz ( 10 msec interal) | | 17500 | | 19200 | | 15900 | | 17600 | |

## D. Consideration of Long Term Objectives

It is important to consider the long term objectives of distributed propulsion control when taking the initial steps toward implementation. The long term approach should emphasize functional modularity and other benefits, such as minimizing the harness length and the number of connectors in the wiring harness. The point-to-point approach discussed in the preceding section is not in alignment with these goals. One reason is that the number of connectors is not reduced. More importantly, however, is that the data flow

on each channel is isolated from the others; minimizing system design flexibility and constraining how functions are grouped within the local nodes. The long term approach should prefer a shared communication media for weight reduction and simplified hardware integration of new subsystems.

There are at least two fundamental decisions to be made about communications in this regard. The first decision considers the sharing of a common communication media by each control element and whether it is event-driven or time-driven in nature. This is discussed extensively in Gwaltny (ref. 3). In an event-driven approach data collisions are a potential result of asynchronous data transfer. Collisions are allowed but result in message termination and retransmission at a later time. Since this is a control application, the possibility of collisions implies that the network is not deterministic and therefore delays in the arrival of information over the network must be accommodated by the control application. Some adjustment for lag is presently used to compensate for variation in system element response. However, increased, potentially unbounded delays place an additional burden on the already demanding control application. Removing system interdependencies is the goal. A slotted or time-triggered protocol, where each control element communicates in a predetermined manner, isolates the control application from the communication function. The penalty, however, may be in reduced communication throughput for a given channel bandwidth because some of the communication "slots" may be unused.

The second decision to be made considers the content of each message, or data exchange and whether it has any dependence on the physical implementation of the control system. In a partially distributed system as shown in figure 4, several control elements, representing multiple control functions, may share a common physical communication interface. But this sharing is strictly a matter of convenience for a given control system hardware implementation. The formatting of the message, i.e., the data communicated by the local control node from or to the sensors and actuators, should not be dependent on the physical implementation of the system. A preferred method would be to allow each system function to communicate as its own virtual entity. In that manner, functions can be easily added to the system as new designs evolve or as capabilities increase without impacting the existing structure. This will enable backward-compatibility as well as accommodation for future growth.

Another advantage of this "virtual" approach to messaging is that it will minimize the message size. Some contrary arguments can be made that larger messages may be more efficient because there is less overhead and less dead time between messages resulting in more effective use of bandwidth. However, large messages are more susceptible to noise and data corruption and require more complicated forward error correction. Large message assembly/disassembly and content would also have to be coordinated between senders and receivers and any changes in control system structure would potentially require modifications to all elements communicating on the shared media, another violation of the functional independence sought after by distributed control.

The decisions about the communication system are important to modular, distributed engine control because the communication system is the key to the development of a standardized open system interface which enables the full realization of the system goals described in section I. Ideally, there should be no difference in the communication structure and requirements between a partially and fully-distributed system. The only differences should be in the physical structure of the control elements gathered around the communication interface. As more high temperature electronics technology is adopted by industry the physical independence and distributed implementation of these control functions should increase.

# V. Fully-Distributed Control System

A fully-distributed control system is characterized by a network of single-function control elements connected to and coordinated by a supervisory controller. In theory, the supervisory controller is another function, or set of functions which could reside in any location. Individually, each control element has a primary identity as a data producer (sensor), or a data consumer (actuator); therefore all loop closures must involve data transfer across the network. One exception to this definition may consider the loop closure around actuator position as integral to the actuator.

## A. Fully-Distributed Strawman Design

The fully-distributed system is an extension of figure 4 into discrete control elements directly connected to the communication system and is shown in figure 5. Local control nodes are not physically separate electronics enclosures, but could be physically integrated into the sensors and actuators themselves. Control loop-closure functions, which are not described, are performed as virtual functions anywhere in the system where the processing capability exists to do so. Separate control law functions might be integrated into the individual actuators or they could continue to reside in a physically separate FADEC.

## B. Loop Closure in the Fully-Distributed System

The restriction imposed by our description of the partially distributed system did not allow local loop-closure unless each variable involved in that decision was under the direct operation of the local node. This limited the closed-loop control to verification of actuator commands using position sensor feedback. Spatially, it was not possible to do more without neglecting the overall goals of the control architecture, such as weight and standardization of interfaces. This restriction was made because the control laws were not independent, i.e., it must be assumed that the operation of any actuator will have an effect within all the controlled variables of the system. The restriction was applied to illustrate the implication of system design decisions on the communication and control architectures.
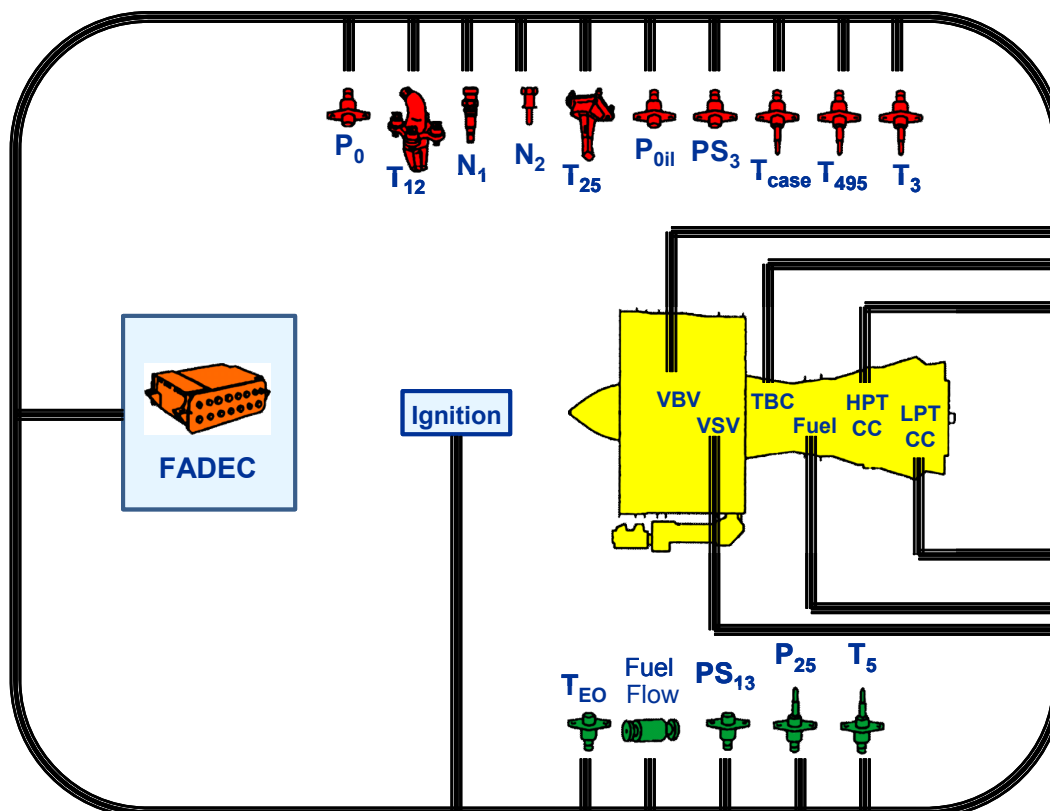


Figure 5.—A fully distributed control system. Each system element individually connects to the network. Each physical element can have multiple functions, some of which require real-time communication for control and others which may be less time critical.

To implement the unrestricted evaluation of control laws anywhere in the system requires a method to account for the shared dependencies between each control loop. Intermediate data, representing these dependencies, are calculated values which result in the partial evaluation of a control law, but are common to more than one loop closure, such as occurs in multivariable control. An intermediate value, if communicated over the network, would appear as sensor-like data but would incorporate a delay since it would be based on past system input data. The potential pitfalls of this approach are that it increases communication system throughput requirements and is potentially unbounded in the number of intermediate values that could be produced. The delays could contribute to system instability (ref. 9). The redundant calculation of these intermediate values, instead of passing them as data between various controllers, requires more processing resources to be provided by the overall system, but is a more modular approach that may lessen the data flow requirements. The redundant calculations could contribute to improved system reliability. While it may not be practical to build systems with this processing capability at present, it is important to note that this flexibility exists in a distributed control system design.

## C. Future Functionality and Data Flow Requirements

Using a common network for communications will increase the need for data throughput from what was previously described in tables I and II. Not only will more data flow over a common channel, due to new system functions being added, but additional overhead in coordinating the information transfer will be required. An assessment of what sort of new information can be expected and how it will affect data flow and communication requirements begins with an examination of the control elements and their functions.

The primary function of a distributed control system sensor is to produce a datum and communicate its value over the network. Table III describes the construction of the sensor datum as a five and one half digit precision value with sign, decimal location, multiplier, and units. The data format is similar to that used by a digital multi-meter display where the digits are encoded as binary coded decimal (BCD) values requiring four bits each. The sign is either plus or minus, requiring 1 b. The half digit is zero or one, requiring one bit. The decimal location is in one of six places, requiring 3 b. The multiplier is 3 b which covers over 18 orders of magnitude using the engineering exponents (milli, micro, nano, etc.). Eight b are reserved to describe the engineering units.

TABLE III.—CONSTRUCTION OF SENSOR DATUM

| bits | sign | decimal | half digit | 5 BCD | multiplier | units | total |
|------|------|---------|------------|-------|------------|-------|-------|
|      | 1    | 3       | 1          | 20    | 3          | 8     | 36    |

The sensor does not need to be concerned with the destination of its information; however it must fully communicate the significance of the information it produces. In real-time control this information would include at least identification of the source of the datum, the datum itself, and check bits for forward error correction. Additional information could include a time stamp although the real-time aspect of the communication structure can insure the relative time of each datum. Valuable information to have in future systems, but not required in real-time, could be device serial number, calibration coefficients, calibration date, and accumulated life. Smart transducers could also incorporate multiple measurements within each sensor function, such as operational temperature, used internally for compensating the sensed parameter. These additional measurements could also be made available to the network.

An intelligent implementation of a sensor could incorporate many additional features. It could accumulate multiple sensor measurements over time in order to provide over-sampling and a variable sliding average of the sensed parameter. It may provide capability for in-situ calibration and execution of built-in test for fault isolation. These, and perhaps many other innovative features, would require what is inherently a data producing device, to communicate in both directions. When being commanded, the sensor must be told who the intended recipient is and where the command originated from.

The primary function of a distributed control system actuator is to produce a mechanical response to a system command received over the network. A commanded device, however, should only respond to valid commands implying it comes from a known source and it is the intended recipient. In real-time control this information would include at least identification of the source, identification of the recipient, the datum, and check bits for forward error correction. An extension to this function could be a command to execute a predefined actuation sequence, although this implies an open-loop type of control. Additional intelligent functions and data requirements for an actuator could include setting and reporting dynamic response coefficients, variable tuning parameters, and calibration parameters, and execution of built-in test for fault isolation. As an output device the actuator could provide serial number, calibration coefficients, calibration date, and accumulated life. Internal measurements of operation temperature, power, and force could also be made available to the network.

Every actuator is assumed to be paired with a feedback position sensor. In a centralized system this loop is closed through the FADEC. In a fully distributed system the loop could be closed through any system controller, however, the value of the position feedback sensor appears to be limited to verifying that the loop was closed. Outside of closing the actuator loop, its value is primarily for health monitoring as opposed to real-time control law execution. In the opening paragraph of section V, the exception for actuator loop closure was intended to question whether a feedback position sensor should be considered as a separate control system function or should every actuator be considered as a data consumer and a data producer as its native implementation, implying that loop closure around an actuator should always be self-contained. This question addresses the specific workings of any control loop and the level of detail which must be reported in real time to the supervisory system.

Future engine control systems will certainly incorporate additional control elements which must be integrated into the communication network (ref. 10). These could include new sensors, actuators, or even complex subsystems for more advanced capabilities. Hopefully a modular, distributed approach to architecture will encourage their inclusion. While it may be difficult to predict the nature of these complex subsystems, it must be noted that the modular nature of the distributed architecture allows many of the details of the internal workings of each control element to be hidden from the larger system, i.e., hidden in the context of real-time data flow. A simple example of this concept is a compressor stall control element in which the dynamic pressure in the compressor is sampled at very high data rates in the tens of kilo-Hertz to detect spike and modal pressure disturbances indicative of compressor instability. By integrating data processing into the control element, the communication over the network may be reduced to an update rate on the order of a few Hertz. Essentially, the communication interface of these systems will serve as a gateway into a separate, localized control network. This technique can effectively limit the real-time need for data flow over the main engine control network.

The preceding assessment suggests that the data flow requirements for a fully-distributed control system can be quite large when considering the range of information that could cross the network. Sensors, normally considered to be data output devices, require information flow in two directions when implemented as intelligent devices. Correspondingly, intelligent actuators could incorporate a multitude of functions which receive or transmit data. New control elements could have almost any level of complexity that could be imagined. However, the major differences in the type of information and its temporal need for transmission on the network can be used to advantage by considering all data to be related to discrete functions instead of the physical control elements within which they are embedded.

## VI. Technical Challenges for Distributed Control Communications

Distributed engine control systems are only possible through the successful design and implementation of a suitable communication system. At a minimum, this communication system must accommodate the real-time data flow requirement for control. However, distributed control will only find acceptance if it enables life-cycle cost reduction features through improved fault isolation made possible with embedded intelligence in system control elements. This intelligence is enabled through the capability afforded by embedded high temperature electronics.

## A. Real-Time Versus Non-Real-Time Data Flow

If one considers the real-time need for data flow across the distributed control network, it is only marginally impacted by its implementation on a shared, digital, serial communication medium. This is because the data that requires the highest real-time throughput is the same data that is currently used to implement control in a centralized system, only the data format has been changed. Table IV shows an estimate of the construction of the real-time functional data flow for the fully distributed control architecture. This rate is constructed using the datum length described in table III as a worst case constant block size message with the additional overhead describing source, destination, and forward error correction. Note that actuators are assumed to be closing the loop internally with an integral position feedback sensor. Even considering future control system advances, the real-time data rate is unlikely to exceed more than double what is estimated in table IV.

TABLE IV.—DATA TRANSMISSION OF REAL-TIME FUNCTIONS
OVER A SINGLE SERIAL NETWORK

| Symbol | datum | source ID | destination ID | forward error correction | total message |
|---|---|---|---|---|---|
| | bits | bits | bits | bits | bits |
| $P_0$ | 36 | 16 | 16 | 7 | 75 |
| $T_{12}$ | 36 | 16 | 16 | 7 | 75 |
| $N_1$ | 36 | 16 | 16 | 7 | 75 |
| $N_2$ | 36 | 16 | 16 | 7 | 75 |
| $T_{25}$ | 36 | 16 | 16 | 7 | 75 |
| $P_{oil}$ | 36 | 16 | 16 | 7 | 75 |
| $PS_3$ | 36 | 16 | 16 | 7 | 75 |
| $T_{case}$ | 36 | 16 | 16 | 7 | 75 |
| $T_{495}$ | 36 | 16 | 16 | 7 | 75 |
| $T_3$ | 36 | 16 | 16 | 7 | 75 |
| $T_{E0}$ | 36 | 16 | 16 | 7 | 75 |
| Fuel Flow | 36 | 16 | 16 | 7 | 75 |
| $PS_{13}$ | 36 | 16 | 16 | 7 | 75 |
| $P_{25}$ | 36 | 16 | 16 | 7 | 75 |
| $T_5$ | 36 | 16 | 16 | 7 | 75 |
| VBV | 36 | 16 | 16 | 7 | 75 |
| VSV | 36 | 16 | 16 | 7 | 75 |
| TBC | 36 | 16 | 16 | 7 | 75 |
| Fuel | 36 | 16 | 16 | 7 | 75 |
| HPTCC | 36 | 16 | 16 | 7 | 75 |
| LPTCC | 36 | 16 | 16 | 7 | 75 |
| Ignition | 36 | 16 | 16 | 7 | 75 |
| Thrust Reverser | 36 | 16 | 16 | 7 | 75 |
| Solenoids | 36 | 16 | 16 | 7 | 75 |
| Total bits per cycle | | | | | 1800 |
| Max. effective bit rate 20 Hz ( 50 msec interval) | | | | | 36000 |
| Max. effective bit rate 100 Hz ( 10 msec interal) | | | | | 180000 |

The more important issue appears to be the accommodation of the multitude of non-real-time, "back-channel," functional data which can flow in the system. It should be expected that this information will continue to increase as new and innovative functions are incorporated in the system. If all this system data were to be continuously transmitted in real-time, within the control law interval time, the data flow requirements of the communication system could be overwhelming and almost impossible to bound.

Various methods could be established to communicate this important, but less than real-time data flow requirement. One possibility is to establish a separate physical communication channel for this lower

rate information. This "back-channel" network could be integrated into the same physical connector to standardize and minimize system impact on the harness and connectors. Another possibility is to segment the data transmission into a combination of fixed and variable data transmission slots. The fixed slots would communicate real-time information for deterministic control using constant and precise time intervals. The variable data transmission slots would function as a virtual "back channel", communicating a variety of system information in round-robin fashion at much reduced data rates. Other communication schemes may exist or could be developed to accommodate this need.

## B. Fault Tolerance

In most systems, redundancy is a primary mechanism for providing fault tolerance. Redundancy is incorporated in all parts of high reliability systems including sensors, actuators, harnessing and computational resources. Distributed systems will also need to incorporate such methods to overcome inevitable faults. Unfortunately, a shared communication medium is a common source of failure for every element in the system. The typical approach in such systems is to provide alternate paths for data to flow in the event one path becomes blocked. There are existing technologies to address these issues which can be incorporated into the system harnessing and the physical interface circuitry used for communications, however, they are beyond the scope of this paper. Their impact on system weight should also be better understood.

Assuming the communication media fails in a distributed system, the best approach may be to physically isolate that segment and rely on the redundant communication channel for control. If a second, similar failure occurs on the redundant communication channel, strategies should be developed to enable crossover communications to maximize system control using the surviving elements of both systems. Recall from Section V that loop-closure in fully distributed systems could be enabled by redundant control law calculations being performed at multiple nodes in the system. This capacity may be used as a form of analytical redundancy to reconstruct a large part of the failed distributed system and circumvent the inherent issues with shared communications networks.

Beyond the limitations of the communication interface, fault isolation is greatly enhanced by embedded intelligence in system elements. The information transmitted over the communication back channel describes embedded functions which can compensate for and/or be used to detect changes in system operation which signal the need for system maintenance before it results in failure.

## C. High Temperature Electronics

As previously stated, the communication media and other embedded electronics are inexorably linked to the capability of implementing their functions in high temperature electronics. Without a capability for circuits operating at elevated temperature, the need for thermal control of electronics becomes necessary. Thermal control adds significant weight and could negate any system benefit from the implementation of distributed control.

Fortunately, there exists a small but increasing capability for electronic components that can be used in the engine environment. Components based on silicon on insulator technology are available that can operate at temperature ranges up to 300 °C. Significant advances in the development of durable, multilevel integration of silicon carbide electronics at temperatures of 500 °C have also recently been achieved (refs. 11 and 12). This bodes well for future engine control capability. It is up to the engine control community to exploit these technologies and develop the market size to support the continued advance of these components.

# VII. Conclusions

The preceding assessment focused on the estimation of data flow requirements in a propulsion engine control application. The assessment began with a given engine control system based on a centralized

architecture. An effective digital data rate was calculated for this analog system based on estimated signal resolution and update rates. The baseline control system was then converted to a partially distributed control system using point-to-point digital communications between local control nodes and a supervisory FADEC. Digital data rates were again estimated based on the additional layer of complexity and a data flow assessment. Finally, the system was converted to a fully distributed engine control system with a shared serial digital communication network. The data flow was again analyzed and the digital communication transmission rates estimated. It was found that the real-time data transmission rates for engine control are increased by the transformation from centralized to distributed architecture. However, these transmission rate increases were primarily based on the data structure and not on an increase in data flow requirements.

Non-real-time data flow may potentially increase to a very large extent based on the increase in embedded functions in the control elements of the system. Technology advance and innovation makes it difficult to put bounds on this increase. Potential ways to address this type of data flow and its impact on communication requirements were discussed. A discussion of fault tolerance was provided which described the system dependency on a shared communication medium. Potential ways to address this issue were discussed.

Finally the dependence of engine control communications on high temperature electronics was discussed. The engine control community must share in the burden of developing systems based on this technology.

This assessment purposely neglected a discussion of existing communication technologies and standards, preferring to focus on the anticipated need of future engine control systems. This approach was taken to avoid the pitfalls of forcing an existing technology to work on an application in lieu of understanding the long term needs of that application. The engine control community needs to assess the validity of the issues raised.

# References

1. Makowitz, R., Temple, C., "Flexray—A Communication Network for Automotive Control Systems," 2006 IEEE International Workshop on Factory Communication Systems, June 27, 2006, pp. 207–212.
2. Lee, K.C., Kim, M.H., Lee, S., Lee, H.H., "IEEE-1451-based Smart Module for In-vehicle Networking Systems of Intelligent Vehicles," *IEEE Transactions on Industrial Electronics*, volume 51, issue 6, Dec. 2004, pp. 1150–1158.
3. Gwaltny, D.A., Briscoe, J.M., "Comparison of Communication Architectures for Spacecraft Modular Avionics Systems," Marshall Space Flight Center, NASA/TM—2006-214431.
4. Culley, D.E., Thomas, R., Saus, J., "Concepts for Distributed Engine Control," 43rd AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit, Cincinnati, OH, July 8–11, 2007, AIAA–2007–5709.
5. Behbahani, A.R, Culley, D., Smith, B.J., et al., "Status, Vision, and Challenges of an Intelligent Distributed Engine Control Architecture," 2007–01–3859, SAE AeroTech Congress & Exhibition, September 2007, Los Angeles, CA.
6. Volponi, A., "Data Fusion for Enhanced Aircraft Engine Prognostics and Health Management," NASA/CR—2005-214055.
7. Soeder, J.F., "F-100 Multivariable Control Synthesis Program—Computer Implementation of the F-100 Multivariable Control Algorithm," NASA TP-2231, 1983.
8. Treager, Irwin E., *Aircraft Gas Turbine Engine Technology*, 3rd ed., Glencoe/McGraw-Hill, 2001, Ch. 3, ISBN 0-02-801831-1.
9. Zhang, W., Branicky, M.S., Phillips, S.M., "Stability of Networked Control Systems," *IEEE Control Systems Magazine*, February 2001,
10. Wills, L., Kannan, S., Sander, S., et al., "An Open Platform for Reconfigurable Control," *IEEE Control Systems Magazine*, June 2001.

11. Neudeck, P.G., Okojie, R.S., Chen, L.-Y, "High-Temperature Electronics—A Role for Wide Bandgap Semiconductors?", *Proceedings of the IEEE*, vol. 90, no. 6, pp. 1065–1076, 2002.
12. Neudeck, P.G., Spry, D.J., Chen, L.Y., et al., "6H-SiC Transistor Integrated Circuits Demonstrating Prolonged Operation at 500 C," IMAPS International Conference and Exhibition on High Temperature Electronics (HiTEC 2008), May 12–15, 2008, Albuquerque, New Mexico.

# REPORT DOCUMENTATION PAGE

*Form Approved*
*OMB No. 0704-0188*

| 1. REPORT DATE *(DD-MM-YYYY)* | 2. REPORT TYPE | 3. DATES COVERED *(From - To)* |
|---|---|---|
| 01-09-2008 | Technical Memorandum | |

| 4. TITLE AND SUBTITLE | 5a. CONTRACT NUMBER |
|---|---|
| Communication Needs Assessment for Distributed Turbine Engine Control | |
| | 5b. GRANT NUMBER |
| | |
| | 5c. PROGRAM ELEMENT NUMBER |
| | |

| 6. AUTHOR(S) | 5d. PROJECT NUMBER |
|---|---|
| Culley, Dennis, E.; Behbahani, Alireza, R. | |
| | 5e. TASK NUMBER |
| | |
| | 5f. WORK UNIT NUMBER |
| | WBS 561581.02.08.03.17.03 |

| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) | 8. PERFORMING ORGANIZATION REPORT NUMBER |
|---|---|
| National Aeronautics and Space Administration<br>John H. Glenn Research Center at Lewis Field<br>Cleveland, Ohio 44135-3191 | E-16583 |

| 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) | 10. SPONSORING/MONITORS ACRONYM(S) |
|---|---|
| National Aeronautics and Space Administration<br>Washington, DC 20546-0001 | NASA |
| | 11. SPONSORING/MONITORING REPORT NUMBER |
| | NASA/TM-2008-215419; AIAA-2007-5281 |

**12. DISTRIBUTION/AVAILABILITY STATEMENT**
Unclassified-Unlimited
Subject Category: 07
Available electronically at http://gltrs.grc.nasa.gov
This publication is available from the NASA Center for AeroSpace Information, 301-621-0390

**13. SUPPLEMENTARY NOTES**

**14. ABSTRACT**
Control system architecture is a major contributor to future propulsion engine performance enhancement and life cycle cost reduction. The control system architecture can be a means to effect net weight reduction in future engine systems, provide a streamlined approach to system design and implementation, and enable new opportunities for performance optimization and increased awareness about system health. The transition from a centralized, point-to-point analog control topology to a modular, networked, distributed system is paramount to extracting these system improvements. However, distributed engine control systems are only possible through the successful design and implementation of a suitable communication system. In a networked system, understanding the data flow between control elements is a fundamental requirement for specifying the communication architecture which, itself, is dependent on the functional capability of electronics in the engine environment. This paper presents an assessment of the communication needs for distributed control using strawman designs and relates how system design decisions relate to overall goals as we progress from the baseline centralized architecture, through partially distributed and fully distributed control systems.

**15. SUBJECT TERMS**
Control; Turbine; Propulsion; Distributed control

| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES | 19a. NAME OF RESPONSIBLE PERSON<br>STI Help Desk (email:help@sti.nasa.gov) |
|---|---|---|---|---|---|
| a. REPORT | b. ABSTRACT | c. THIS PAGE | UU | 24 | 19b. TELEPHONE NUMBER *(include area code)* |
| U | U | U | | | 301-621-0390 |

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39-18